

## AMENDMENTS TO THE SPECIFICATION

### PLEASE AMEND THE SPECIFICATION AS FOLLOWS:

Please replace the last paragraph on page 12 with the following amended paragraph:

In this example, the contents of the `<x:test>` section is evaluated. The result is a document fragment containing the elements `<actual>` and `<submitted>` with some values inside them. The local path in the “x:select” attribute on the ~~`<x:text>`~~ `<x:test>` element is executed in the context of the document fragment result of the test. The result of the path is casted to a path Boolean. Lastly, if the Boolean result was true, the `<x:then>` section is ~~eevaluated~~ evaluated and its contents become the value of the `<x:if>`. If the Boolean result was false, the `<x:else>` section is evaluated and its contents become the value of the `<x:if>`.

Please replace the last paragraph on page 13 with the following amended paragraph:

In this example, the contents of the ~~`<x:eah>`~~ `<x:each>` section is evaluated first. If a “x:select” attribute is present, it is evaluated and its result is interpreted as a node *list* (i.e. not a single node). In this case, the node list has two nodes: the two `<record>` elements. The name “\$rec” is bound to each of the nodes in the list, beginning with the first. If the name shadows other name that is in scope, it is an error. That is, if there is a `<x:xcell>` names “rec” or and out `<x:for>` using a variable called “rec”, an error will be signaled. If no error, for each of nodes in the list, the contents of the `<x:do>` section is evaluated once. The value of the `<x:for>` if the document fragment containing the concatenated values of all the evaluated `<x:do>` sections. Outside the `<x:for>`, the variable `<$rec>` is meaningless, and any reference would be signaled as an error.